基于潜在标签挖掘和细粒度偏好的个性化标签推荐 *

李红梅 [1]，刁兴春 [1]，曹建军 [2†]，张　磊 [1]，冯　钦 [1]

(1. 陆军工程大学，南京 210007; 2. 国防科技大学 第六十三研究所，南京 210007)

摘　要：为进一步提高个性化标签推荐性能，针对标签数据的稀疏性以及传统方法忽略隐藏在用户和项目上下文中潜在标签的缺陷，提出一种基于潜在标签挖掘和细粒度偏好的个性化标签推荐方法。首先，提出利用用户和项目的上下文信息从大量未观测标签中挖掘用户可能感兴趣的少量潜在标签，将标签重新划分为正类标签、潜在标签和负类标签三类，进而构建<用户，项目>对标签的细粒度偏好关系，在缓解标签稀疏性的同时，提高对标签偏好关系的表达能力；然后，基于贝叶斯个性化排序优化框架对细粒度偏好关系进行建模，并结合成对交互张量分解对偏好值进行预测，构建细粒度的个性化标签推荐模型并提出优化算法。对比实验表明，提出的方法在保证较快收敛速度的前提下，有效地提高了个性化标签的推荐准确性。

关键词：个性化标签推荐；潜在标签挖掘；贝叶斯个性化排序；成对交互张量分解

中图分类号：TP301.6　　doi: 10.19734/j.issn.1001-3695.2018.05.0498

Personalized tag recommendation based on potential tag mining and fine-grained preference

Li Hongmei[1], Diao Xingchun[1], Cao Jianjun[2†], Zhang Lei[1], Feng Qin[1]

(1. *Army Engineering University*, *Nanjing* 210007, *China*; 2. *the* 63*rd Research Institute*, *National University of Defense Technology*, *Nanjing* 210007, *China*)

Abstract: To further improve the performance of personalized tag recommendation, this paper argued that traditional methods ignore the potential and informative tags hidden in the context of users and items. Aimed at this, this paper proposed a novel personalized tag recommendation method BPR-PITF-P based on potential tag mining and fine-grained preference. Firstly, BPR-PITF-P leverages the context information of both users and items to mine potential and useful tags, and gets three kinds of tags: positive tags, potential tags, and negative tags. Based on the above, it translates the traditional pairwise preference into fine-grained preference relationship among user-item post and tags. This kind of treatment helps alleviate the sparse problem of tagging data. Second, combined with pairwise interaction tensor factorization method to predict preference value, BPR-PITF-P models the preference relationship based on the optimization criteria of Bayesian personalized ranking, and develops a personalized tag recommendation model followed by optimization algorithm. The comparison results show that our proposed method could improve tag recommendation performance in the premise of guarantee convergence speed.

Key words: : tag recommendation; potential tag mining; Bayesian personalized ranking; pairwise interaction tensor factorization

0　引言

作为 Web 2.0 的重要特征，社会标签允许用户自由创建标签对资源（如网页、音乐、电影、图片等，统称项目）进行标注。标签既描述了项目的显式语义特征，又反映了用户的显式偏好内容，使得用户能够更加便捷地检索、组织、分享信息等，在一定程度上缓解了“信息过载”的问题。

标签系统非常希望用户能够为资源打上高质量的标签，促进对资源的分享与利用。很多网站如 Lastfm、MovieLens、Delicious、CiteULike、豆瓣网等都设计了标签推荐模块给用户推荐标签，这样用户在浏览资源时，标签推荐系统提供一些用户可能感兴趣的个性化标签，以便其能够更好地标注和管理这些资源。但由于有许多用户懒于为信息资源添加标签，所以需要个性化标签推荐系统自动为其推荐与信息资源相关的标签。

典型的个性化标签推荐方法主要包括基于协同过滤的标签推荐算法[1]、基于图模型的标签推荐算法[1~3]、基于张量分解的标签推荐[4~7]、基于内容的标签推荐算法[8~10]和混合的标签推荐算法[11]等，这些方法各有优缺点，适应不同的应用场景。另外，

收稿日期：2018-05-08；**修回日期：**2018-06-23　**基金项目：**国家自然科学基金资助项目（61371196）；中国博士后科学基金资助项目（2015M582832）

作者简介：李红梅（1990-），女，博士研究生，主要研究方向为个性化推荐；刁兴春（1964-），男，研究员，博导，主要研究方向为数据工程；曹建军（1975-），男（通信作者），副研究员，硕导，主要研究方向为数据智能分析与应用、数据质量控制与数据治理（jianjuncao@yeah.net）；冯钦（1993-），男，硕士研究生，主要研究方向为数据工程；张磊（1989-），男，硕士研究生，主要研究方向为数据工程.

也有一些文献提出利用额外的上下文信息（如时间[12,13]、社交关系[14]等）来辅助提高个性化标签推荐的能力。由于标签推荐系统中的数据通常较为稀疏，严重影响着推荐算法的性能,而张量分解方法能有效地处理稀疏数据，具有准确的预测性、良好的扩展性等优点，所以本文主要关注基于张量分解的标签推荐算法。

用于标签推荐的典型的张量分解方法包括高阶奇异值分解（higher-order-singular-value-decomposition，HOSVD）[15]、排序张量分解（ranking tensor factorization，RTF）[4]，这些方法都是基于经典的 Tucker 分解模型（tucker decomposition，TD），但其计算复杂度比较高[5]。为进一步降低计算复杂度，文献[5]提出一种基于成对交互张量分解（pairwise interaction tensor factorization，PITF）的方法，同时与当前较为流行的、基于成对偏好假设的贝叶斯个性化排序（Bayesian personalized ranking，BPR）[16]方法相结合，利用<用户,项目>对标签的成对偏好关系进行建模，在提高计算效率的同时，提高了推荐的准确性。这是因为针对某个项目，用户通常主要关注少数排在前面的、符合用户偏好的标签，基于排序学习的 BPR 方法能够有效地对该偏好进行建模与优化。该方法认为针对某个项目，用户对添加的标签（即观测标签）的偏好大于未添加的标签（即未观测标签），并使用均匀抽样的方法来抽取成对标签。而标签数据的长尾分布特性使得均匀抽样方法存在一定的缺陷：一方面抽取一些没有意义的标签对，影响模型的推荐准确性，另一方面不能提供有效信息而导致模型收敛速度过慢。针对该问题，文献[7]在文献[5]的基础上进一步提出了基于自适应、过采样的成对抽样方法，来提高模型的收敛能力与标签推荐性能。但该方法过于关注数据长尾分布中的流行标签，忽略部分隐藏在长尾数据中潜在、有意义的标签，容易导致过早收敛。同时，上述方法受稀疏性的影响，忽略了用户和项目的上下文信息对不同标签偏好关系的影响。

由于未观测标签的规模通常远远大于观测标签的规模，数据的长尾分布容易导致抽样的偏置，且忽略了标签与用户和项目的关联关系等信息，将不同标签同等看待，降低对偏好关系的表达能力。若能从大量未观测标签中挖掘出少量用户可能感兴趣的潜在标签，构建具有可比性的、更加有意义的标签对，可在一定程度上缓解稀疏性的影响，同时提高<用户，项目>对标签的偏好表达能力及模型的推荐准确性。事实上，这些潜在标签可通过目标用户和项目的上下文中获取，一是其他用户对该项目添加的标签，这些标签很好地表征项目的属性，二是该用户为其他类似项目添加的标签，这些标签则表达用户的偏好内容。相对于其他未观测标签，这些潜在标签与目标用户和项目关联更加紧密，更加可能为用户所感兴趣，有必要将其与其他未观测标签区别对待，进而构建更加具有可比性的标签对，为排序推荐模型提供有效的信息。

基于以上分析，提出一种基于潜在标签挖掘的个性化标签推荐方法。利用用户和项目的上下文信息从未观测标签中挖掘少数潜在的标签，重新定义三类标签：正类标签、潜在标签和负类标签，一方面可缓解标签数据的稀疏性，另一方面提高<用户，项目>对不同类型标签的偏好能力的表达。这样，基于三类标签提出一种细粒度偏好关系，改进传统方法中<用户，项目>对标签的成对偏好，并利用贝叶斯个性化排序准则来优化偏好关系，提高个性化标签推荐性能。

1　潜在标签挖掘与细粒度偏好关系构建

首先，利用用户和项目的上下文关系挖掘潜在、有价值的标签样本，构建三类标签：正类标签、潜在标签、负类标签；然后，基于样本重新构建（用户，项目）对不同类型标签的偏好关系。

1.1　潜在标签挖掘

由于未观测标签的集合规模相对较大，从未观测标签集合中抽取更加有价值的样本，对提高推荐模型的性能起着重要作用。如何挖掘潜在有用的标签？本文认为可利用用户和项目的上下文信息来挖掘一些潜在的、有价值的标签，重新构建样本集及<用户,项目>对标签的偏好关系。

下面举例说明。表 1 给出数据集中的部分<用户，项目，标签>三元组样例，个性化标签推荐的目的是为某个用户针对某个项目推荐可能感兴趣的标签，即为某个<用户，项目>组合推荐 top-*n* 标签列表。

表 1　用户-项目-标签)三元组样例

Table 1　user-iten-lable) triple sample

用户	项目	标签
u_1	i_1	t_3
u_1	i_1	t_4
u_2	i_1	t_5
u_2	i_1	t_6
u_1	i_3	t_7
u_1	i_3	t_8
u_2	i_4	t_9
u_2	i_4	t_{10}

如表 1 所列，用户 u_1 为项目 i_1 添加了标签 t_3, t_4。对于 (u_1,i_1) 而言，传统的标签推荐方法认为 (u_1,i_1) 对观测标签 t_3, t_4 的偏好程度大于其他未观测标签 $t_5, t_6, t_7, t_8, t_9, t_{10}$，甚至将标签 $t_5, t_6, t_7, t_8, t_9, t_{10}$ 同等对待。然而，这些方法忽略了用户和项目的上下文关系对标签推荐的影响。

例如，项目 i_1 除了被用户 u_1 标记为标签 t_3, t_4 外，也被其他用户 u_2 添加了标签 t_5, t_6，表明对于项目 i_1 而言，标签 t_5, t_6 也能很好地表达其特征，标签 t_5, t_6 可构成用户感兴趣的潜在标签。同样地，用户 u_1 除了对标签 t_3, t_4 感兴趣外，也对标签 t_7, t_8 感兴趣，标签 t_7, t_8 可构成用户感兴趣的潜在标签。因此，可认为 (u_1,i_1) 对未观测标签 t_5, t_6, t_7, t_8 的偏好程度可能大于其他未观测标签 t_9, t_{10}，可将标签 t_5, t_6, t_7, t_8 视为 (u_1,i_1) 潜在感兴趣的标签，利用这些潜在的标签可更好地表达<用户，项目>对标签的偏好

关系，为个性化标签推荐提供有价值的信息，进而提高推荐模型的性能。

事实上，本文提出的思想类似于文献[17]和文献[18]，这些文献通过利用用户社会关系等挖掘用户可能更加感兴趣的潜在项目，来构建用户对项目的细粒度偏好关系，进而构建更加有效的模型提升对项目个性化推荐性能，这些方法为本文的思路提供有力的支撑。但针对个性化标签推荐的任务，当缺乏用户社会关系时，如何利用仅有的用户-项目-标签交互数据来挖掘潜在的标签是一大关键步骤。本文则提出了充分利用用户和项目的上下文来实现潜在标签的挖掘，从大量未观测标签数据中挖掘规模较小的、与目标用户和项目紧密关联的潜在标签，形成不同类型的标签，进而构建<用户，项目>对标签的偏好关系。

1.2 细粒度偏好关系

基于前面构建的三类标签，提出将传统的成对偏好关系假设：观测标签 $\succ_{u_1,i_1}$ 未观测标签，改进为细粒度的偏好关系假设：观测标签 $\succ_{u_1,i_1}$ 潜在标签 $\cap$ 观测标签 $\succ_{u_1,i_1}$ 其他未观测标签。这种细粒度偏好关系可形式化为 $t_A \succ_{u_1,i_1} t_p \cap t_A \succ_{u_1,i_1} t_B$，其中 t_A 为观测标签 $t_A \in \{t_3, t_4\}$，t_p 为潜在标签 $t_p \in \{t_5, t_6, t_7, t_8\}$，$t_B$ 为其他未观测标签 $t_B \in \{t_9, t_{10}\}$。相对于传统的成对偏好关系，本文提出的细粒度偏好关系能充分挖掘用户潜在的偏好，提高用户偏好关系的表达能力。

根据以上假设，为每个 (u,i) 形式化定义三类标签：正类标签 T_A、潜在标签 T_P 和负类标签 T_B。其中，正类标签即出现在数据集中的观测标签，潜在标签和负类标签构成非观测标签。

1）正类标签 T_A：$T_A = \{t_A \mid (u,i,t_A) \in S\}$

2）潜在标签 T_P：$T_P = \{t_p \mid (u,i,t_p) \in S_P\}$

3）负类标签 T_B：$T_B = \{t_B \mid (u,i,t_B) \notin S \cap (u,i,t_B) \notin S_P\}$

其中，S 为训练集，包含大量（用户，项目，标签）三元组，$S = \{(u,i,t) \mid u \in U, i \in I, t \in T\} \subseteq U \times I \times T$，$U$ 为用户集合，I 为项目集合，T 为标签集合。$S_P = T_i \cup T_u$，具体地，从项目上下文的角度，可挖掘潜在标签 T_i：$T_i = \{t \mid (i,t) \subset S\}$。从用户上下文的角度，可挖掘潜在标签 T_u：$T_u = \{t \mid (u,t) \subset S\}$。这样，对于 (u,i) 而言，可挖掘潜在标签 $S_P = T_i \cup T_u$。

基于三类标签数据，构建用户的偏好关系，形式化为 $t_A \succ_{u,i} t_p \cap t_A \succ_{u,i} t_B$，其中 $t_A \in T_A$，$t_p \in T_P$，$t_p \in T_P$。基于该偏好关系，便可利用 BPR-PITF 模型[5]对偏好关系进行建模与求解。

2 个性化标签推荐模型与求解

2.1 个性化标签推荐模型

为<用户，项目>推荐其可能感兴趣的标签，本质上是一种面向排序的任务，尽可能将用户最可能感兴趣的项目排在前面，从而提高用户的满意度。贝叶斯个性化排序 BPR 是一种基于排序学习的个性化推荐优化准则与框架。成对交互张量分解 PITF 则是用于计算<用户，项目>对某个标签的偏好值的经典方法，类似于常用的矩阵分解模型，PITF 具有较好的理论基础、良好的扩展性、预测的准确性等优点，可与 BPR 优化框架进行很好地结合进行标签推荐。下面主要介绍 BPR-PITF 方法。

传统的 BPR 建立在<用户 u，项目 i>对标签成对偏好假设的基础上，其认为 (u,i) 对观测标签 t_A 的偏好大于未观测标签 t_B，即 $t_A \succ_{u,i} t_B$，目的是尽可能让观测标签排在未观测标签前面。基于贝叶斯理论，BPR 通过最大化排序目标函数的后验概率（式（1）来获取项目的正确排序

$$p(\Theta \mid \succ_{u,i}) \propto p(\succ_{u,i} \mid \Theta)\, p(\Theta) \tag{1}$$

其中，$p(\succ_u \mid \Theta)$ 表示 (u,i) 关于标签成对排序的似然函数。$p(\Theta)$ 表示参数的先验概率函数，Θ 表示 PITF 模型的参数。

假设不同 (u,i) 的偏好相互独立，最大化后验概率 MAP 等效于

$$\max \prod_{(u,i)\in S} p(t_A \succ_{u,i} t_B \mid \Theta)\, p(\Theta) \tag{2}$$

其中，S 表示训练集中的<用户，项目，标签>三元组集合 $S = \{(u,i,t_A) \mid u \in U, i \in I, t_A \in T\}$，$t_A$ 为出现在训练集中的观测标签，t_B 则为未出现在训练集中的未观测标签。$p(t_A \succ_u t_B \mid \Theta)$ 表示 (u,i) 对标签 t_A 和 t_B 的偏序关系的概率，常用 sigmoid 函数来表达

$$p(t_A \succ_{u,i} t_B \mid \Theta) = \sigma(\hat{F}_{u,i,t_A}(\Theta) - \hat{F}_{u,j,t_B}(\Theta)) \tag{3}$$

其中: $\sigma(x) = 1/(1+\exp(-x))$。$\hat{F}_{u,i,t_A}(\Theta)$ 为 (u,i) 对标签 t_A 的预测评分，采用 PITF 模型来计算。

PITF 是张量分解的一种特殊形式，利用用户、项目、标签三者之间的两两交互关系来对偏好进行建模，通过对三种交互关系分解来计算 (u,i) 对标签 t 的偏好值

$$\hat{F}_{u,i,t_A}(\Theta) = U_u \cdot V_i + U_u \cdot T^U_{t_A} + V_i \cdot T^V_{t_A} \tag{4}$$

对应的参数 Θ 为用户、项目和标签的隐因子矩阵 $U \in \mathbb{R}^{M\times K}$、$V \in \mathbb{R}^{N\times K}$、$T^U \in \mathbb{R}^{P\times K}$、$T^V \in \mathbb{R}^{P\times K}$。其中，$M$ 为用户个数，N 为项目个数，P 为标签个数。对于先验概率 $p(\Theta)$，常用均值为 0 的高斯分布函数来表示。

这样，通过最小化负对数后验概率，得到 BPR-PITF 的目标函数

$$\min_{\Theta=\{U,V,T^U,T^V\}} L_1 = -\sum_{(u,i,t_A)\in S}\sum_{t_B\in T_B} \ln\sigma(\hat{F}_{u,i,t_A}(\Theta) - \hat{F}_{u,j,t_B}(\Theta)) + \frac{\lambda}{2}\|\Theta\|_2^2 \tag{5}$$

其中：λ 为正则项因子，用于防止过拟合，$\|\Theta\|_2^2 = \|U\|_2^2 + \|V\|_2^2 + \|T^U\|_2^2 + \|T^V\|_2^2$。

针对基于潜在标签挖掘获得的细粒度用户偏好关系 $t_A \succ_{u,i} t_p \cap t_A \succ_{u,i} t_B$，利用 BPR-PITF 优化框架构建新的个性化标签推荐模型，其目标函数为

$$\min_{\Theta=\{U,V,T^U,T^V\}} L_2 = -\sum_{(u,i,t_A)\in S}\sum_{t_P\in T_P} \ln\sigma(\hat{F}_{u,i,t_A}(\Theta) - \hat{F}_{u,j,t_p}(\Theta)) - \sum_{(u,i,t_A)\in S}\sum_{t_B\in T_B} \ln\sigma(\hat{F}_{u,i,t_A}(\Theta) - \hat{F}_{u,j,t_B}(\Theta)) + \frac{\lambda}{2}\|\Theta\|^2 \tag{6}$$

由于提出的模型基于 BPR-PITF 和潜在标签（Potential tags）进行个性化标签推荐，因此又称该模型为 BPR-PITF-P。

2.2　模型求解

针对每个样本 (u,i,t_A,t_P,t_B)，采用随机梯度下降法（SGD）迭代更新参数 $\Theta=\{U,V,T^U,T^V\}$，学习率为 η 。

参数 $\Theta=\{U,V,T^U,T^V\}$ 的梯度可表示为

$$\frac{\partial L_2}{\partial\Theta}=-(1-\sigma(\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_P}(\Theta)))\frac{\partial(\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_P}(\Theta))}{\partial\Theta}-(1-\sigma(\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_B}(\Theta)))\frac{\partial(\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_B}(\Theta))}{\partial\Theta}+\lambda\Theta$$

其中:

$$\begin{aligned}&\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_P}(\Theta)\\&=(U_u\cdot V_i+U_u\cdot T^U_{t_A}+V_i\cdot T^V_{t_A})-(U_u\cdot V_i+U_u\cdot T^U_{t_P}+V_i\cdot T^V_{t_P})\\&=U_u\cdot(T^U_{t_A}-T^U_{t_P})+V_i\cdot(T^V_{t_A}-T^V_{t_P})\end{aligned}$$

分别令

$$\delta_{u,i,t_A,t_P}=-(1-\sigma(\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_P}(\Theta)))\tag{7}$$

$$\delta_{u,i,t_A,t_B}=-(1-\sigma(\hat{F}_{u,i,t_A}(\Theta)-\hat{F}_{u,j,t_B}(\Theta)))\tag{8}$$

具体地，

1）对 U 更新

$$\frac{\partial L_2}{\partial U_u}=\delta_{u,i,t_A,t_P}(T^U_{t_A}-T^U_{t_P})+\delta_{u,i,t_A,t_B}(T^U_{t_A}-T^U_{t_B})+\lambda U_u\tag{9}$$

$$U_u=U_u-\eta\cdot\frac{\partial L_2}{\partial U_u}\tag{10}$$

2）对 V 更新：

$$\frac{\partial L_2}{\partial V_i}=\delta_{u,i,t_A,t_P}(T^V_{t_A}-T^V_{t_P})+\delta_{u,i,t_A,t_B}(T^V_{t_A}-T^V_{t_B})+\lambda V_i\tag{11}$$

$$V_i=V_i-\eta\cdot\frac{\partial L_2}{\partial V_i}\tag{12}$$

3）对 T^U 更新：

$$\frac{\partial L_2}{\partial T^U_{t_A}}=\delta_{u,i,t_A,t_P}U_u+\delta_{u,i,t_A,t_B}U_u+\lambda T^U_{t_A}\tag{13}$$

$$T^U_{t_A}=T^U_{t_A}-\eta\cdot\frac{\partial L_2}{\partial T^U_{t_A}}\tag{14}$$

$$\frac{\partial L_2}{\partial T^U_{t_P}}=\delta_{u,i,t_A,t_P}(-U_u)+\lambda T^U_{t_P}\tag{15}$$

$$T^U_{t_P}=T^U_{t_P}-\eta\cdot\frac{\partial L_2}{\partial T^U_{t_P}}\tag{16}$$

$$\frac{\partial L_2}{\partial T^U_{t_B}}=\delta_{u,i,t_A,t_B}(-U_u)+\lambda T^U_{t_B}\tag{17}$$

$$T^U_{t_B}=T^U_{t_B}-\eta\cdot\frac{\partial L_2}{\partial T^U_{t_B}}\tag{18}$$

4）对 T^V 更新：

$$\frac{\partial L_2}{\partial T^V_{t_A}}=\delta_{u,i,t_A,t_P}V_i+\delta_{u,i,t_A,t_B}V_i+\lambda T^V_{t_A}\tag{19}$$

$$T^V_{t_A}=T^V_{t_A}-\eta\cdot\frac{\partial L_2}{\partial T^V_{t_A}}\tag{20}$$

$$\frac{\partial L_2}{\partial T^V_{t_P}}=\delta_{u,i,t_A,t_P}(-V_i)+\lambda T^V_{t_P}\tag{21}$$

$$T^V_{t_P}=T^V_{t_P}-\eta\cdot\frac{\partial L_2}{\partial T^V_{t_P}}\tag{22}$$

$$\frac{\partial L_2}{\partial T^V_{t_B}}=\delta_{u,i,t_A,t_B}(-V_i)+\lambda T^V_{t_B}\tag{23}$$

$$T^V_{t_B}=T^V_{t_B}-\eta\cdot\frac{\partial L_2}{\partial T^V_{t_B}}\tag{24}$$

2.3　BPR-PITF 算法

BPR-PITF-P 算法

1. 输入：训练集 S，每个 (u,i) 的潜在标签集合 T_P、负类标签集合 T_B，学习率 η，最大迭代次数 κ
2. 输出：参数 $\Theta=\{U,V,W\}$
3. 初始化　$\Theta\sim N(0,0.01)$
4. repeat
5. 　从训练集 S 中均匀抽取（用户，项目，标签）三元组 (u,i,t_A)
6. 　从潜在标签集合 T_P 中随机抽取潜在标签 t_P
7. 　从负类标签集合 T_B 中随机抽取负类标签 t_B
8. 　计算 δ_{u,i,t_A,t_P}（式 7）、δ_{u,i,t_A,t_B}（式 8）
9. 　更新 U：式（9）-式（10）
10. 　更新 V：式（11）-式（12）
11. 　更新 T^U：式（13）-式（18）
12. 　更新 T^V：式（19）-式（24）
13. until 目标函数（式 6）未收敛或者迭代次数≥ κ
14. return　$\Theta=\{U,V,T^U,T^V\}$

第 5～7 行进行样本的抽样，得到<用户，项目，正类标签，潜在标签，负类标签>五元组。具体地，针对训练集 S 中的每个三元组<用户，项目，正类标签>(u,i,t_A)，从每个 (u,i) 的潜在标签集合 T_P 中随机抽取潜在标签 t_P，从每个 (u,i) 的负类标签集合 T_B 中随机抽取负类标签 t_B，进而构建样本 (u,i,t_A,t_P,t_B)。

第 8～12 行分别计算模型关于参数 $\Theta=\{U,V,T^U,T^V\}$ 的梯度，并利用随机梯度下降法对参数进行更新。利用式（9）（10）更新参数 U，利用式（11）（12）更新参数 V，利用式（13）～（18）更新参数 T^U，利用式（19）～（24）更新参数 T^V。

值得注意的是，根据文献[7]，通常每次迭代涉及 $|S|$ 个 BPR 更新过程，其中 $|S|$ 训练集中 (u,i,t_A) 的个数，每个 BPR 更新基于一个样本 (u,i,t_A,t_P,t_B) 进行，这样每次迭代即可遍历一遍训练集 S，保证每次迭代过程中所有观测样本都能参与训练。

第 13 行判断迭代是否收敛；第 14 行返回参数 $\Theta=\{U,V,T^U,T^V\}$。

训练得到参数后，便可根据式（4）$\hat{F}_{u,i,t_A}=U_u\cdot V_i+U_u\cdot T^U_t+U_u\cdot T^V_t$ 计算测试集中（用户，项目）对所有未观测标签的评分，根据评分大小对标签进行排序，得到 top-n 个性化标签推荐列表。

复杂度分析：算法每次迭代涉及 $|S|$ 个样本，针对每个样本

(u,i,t_A,t_P,t_B) 对参数 $\Theta=\{U,V,T^U,T^V\}$ 进行更新的复杂度主要与隐特征维度 K 相关，因此每次迭代的时间复杂度为 $O(K)$。另外，由于样本中潜在标签 t_P 和负类标签 t_B 的确定可在训练前进行预处理，不影响整个模型的训练时间。

3 实验结果与分析

3.1 数据集

实验采用两个数据集进行训练与测试：Lastfm 和 Movielens 数据集（https://grouplens.org/datasets/hetrec-2011），这两个数据集是用于推荐领域的标准数据集。

a）Lastfm：Lastfm 是国外流行的主要包含用户对音乐网站 Last.fm 中歌手的收听、标签等信息。数据集是以多个<用户，项目，标签>三元组的形式出现。该数据非常稀疏，首先根据文献[5]的 p-core 方法对数据进行预处理（p=10），保证每个用户、项目、标签至少在 10 个<用户，项目>对中出现。预处理后的数据集包括 614 用户、1715 个歌手、873 个标签，总共 87285 条记录。

b）MovieLens：MovieLens 是电影推荐网站，允许用户为电影添加标签，以方便电影的搜索与推荐。首先对其进行预处理根据文献[5]的 p-core 方法对数据进行预处理（p=10），保证每个用户、项目、标签至少在 5 个<用户，项目>对中出现。预处理后的数据集包括 366 用户、1185 个歌手、873 个标签，总共 20089 条记录。

实验采用 5-交叉验证的方法训练与测试模型，并取平均值作为最终结果。根据文献[5]构建训练集和测试集，数据集中存在多个<用户，项目>对，每个用户存在于多个<用户，项目>对中。针对数据集中出现的每个用户，抽取一个<用户，项目>对及其相关标签，构成测试集 S_{test}，其余即为训练集 S_{train}。

3.2 评价标准

利用准确率 Precision@n、召回率 Recall@n 和 F1@n 对个性化标签推荐的结果进行评价[5]。

$$\text{Precision}@n=\underset{(u,i)\in PS_{test}}{avg}\frac{Top(u,i,n)\cap\{t\mid(u,i,t)\in S_{test}\}}{n}$$

$$\text{Recall}@n=\underset{(u,i)\in PS_{test}}{avg}\frac{Top(u,i,n)\cap\{t\mid(u,i,t)\in S_{test}\}}{|\{t\mid(u,i,t)\in S_{test}\}|}$$

$$\text{F1}@n=\frac{2\cdot\text{Precision}@n\cdot\text{Recall}@n}{\text{Precision}@n+\text{Recall}@n}$$

其中，PS_{test} 表示训练集 S_{test} 中出现的所有<用户，项目>对。$Top(u,i,n)$ 表示针对 (u,i)，在标签排序列表中出现在前 n 的标签集合，$\{t\mid(u,i,t)\in S_{test}\}$ 测试集中与 (u,i) 相关的标签集合。

3.3 对比方法与参数设置

实验与当前较为流行的标签推荐方法进行对比，包括：

a）Pop。该方法基于标签的流行度进行推荐，主要针对每个项目统计所有标签被标注的次数，并推荐标注次数较高的标签。该方法作为一种基准，用于验证其他方法的有效性。

b）BPR-PITF-U[5]。该方法基于 BPR-PITF 优化准则构建个性化标签推荐的目标函数，采用均匀抽样的方法来训练模型。

c）BPR-PITF-A[7]。该方法基于 BPR-PITF 优化准则构建个性化标签推荐的目标函数，基于指数分布进行自适应抽样来训练模型。

d）BPR-PITF-P。本文提出的方法，首先抽取潜在标签构建训练样本，形成新的偏好关系，并基于 BPR-PITF 优化准则和偏好关系重新构建个性化标签推荐模型。

参数分别 $\Theta=\{U,V,T^U,T^V\}$ 初始化为均值为 0、方差为 0.01 的标准正态分布。隐特征矩阵的维数分别设定为 $d=\{20,40\}$，学习率 $\eta=0.025$，最大迭代次数 $\kappa=1000$。令 $n=\{1,2,\ldots,10\}$，测试所有方法在不同 top-n 下的个性化标签推荐性能。对于正则项系数 λ，针对每个数据集选择最优的 λ（Lastfm 数据集 $\lambda=0.01$，Movielens 数据集 $\lambda=0.01$）

3.4 实验结果

3.4.1 Top-n 推荐性能对比

如图 1 和图 2 所示，在 Lastfm 数据集和 Movielens 数据集上，本文提出的方法 BPR-PITF-P 在三个评价指标上均表现最优，证明本文提出方法能有效地提高个性化标签推荐的准确性。

具体地，在 Lastfm 数据集上，当 K=20 时（见图 1（a）、（c）、（e）），Pop、BPR-PITF-U、BPR-PITF-A、BPR-PITF-P 四个方法的准确率 Precision@n 随着 top n 的增大则降低，召回率 Recall@n 随着 top n 的增大则增大，F1 值 F1@n 则随着 top n 的增大先增大后减小，并在 $n=4$ 处取得最优值。当 K=40 时（见图 1（b）、（d）、（f）），BPR-PITF-U、BPR-PITF-A、BPR-PITF-P 三个方法表现出类似的推荐性能，但相对于 K=20，K=40 时各个方法的推荐性能并没有得到太多的提升。这是因为对于隐特征分解模型来说，通常少数的隐特征足以表达用户、项目、标签的特征，这样能保证在可接受的时间内取得足够好的推荐性能。由于 Pop 方法不受隐特征维度 K 的影响，其分别在 K=20 和 K=40 时性能一致。

相比其他方法，Pop 方法表现最差。这是因为 Pop 方法仅考虑用于标注项目的最流行的标签，未考虑用户的个性化需求，显然 Pop 这种非个性化标签推荐方法的性能差于个性化标签推荐方法（包括其他对比方法 BPR-PITF-U、BPR-PITF-A、以及本文提出的方法 BPR-PITF-P）的性能。

在 Movielens 数据集上（见图 2），Pop、BPR-PITF-U、BPR-PITF-A、BPR-PITF-P 四个方法在三个评价指标上表现出与 Lastfm 数据集类似的结果。不同的是，Movielens 数据集的 F1@n 值在 $n=2$ 处取得最优值，且随着 n 的增大，F1@n 值迅速降低。这是因为 Movielens 数据集的稀疏性大于 Lastfm 数据集，对于大多数（用户，项目）来说，其被标注的标签较少，因此当 n 为较小值时，即 $n=2$，Movielens 数据集取得较好的结果。

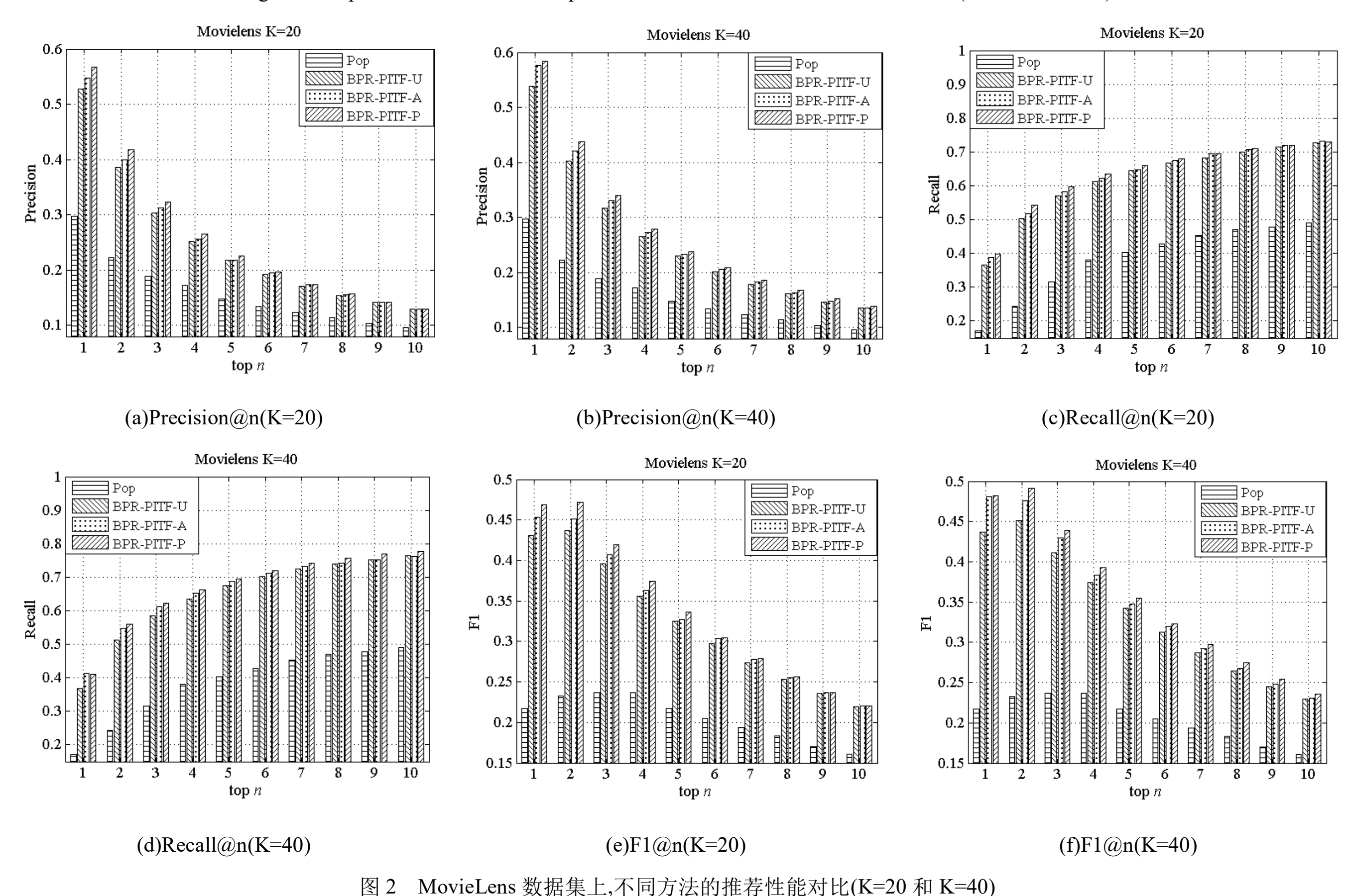
因此，在下节中，为比较不同方法的收敛性能，在 Lastfm 数据集上仅关注 $n=4$ 时的 top-n 推荐性能随迭代次数的变化趋势，在 Movielens 数据集上仅关注 top-2 推荐性能。

(a)Precision@n(K=20)　　(b)Precision@n(K=40)　　(c)Recall@n(K=20)

(d)Recall@n(K=40)　　(e)F1@n(K=20)　　(f)F1@n(K=40)

图 1　Lastfm 数据集上,不同方法的推荐性能对比

Fig.1　Comparison of recommended performance of different methods on Lastfm (K=20 and K=40)



(a)Precision@n(K=20)　　(b)Precision@n(K=40)　　(c)Recall@n(K=20)

(d)Recall@n(K=40)　　(e)F1@n(K=20)　　(f)F1@n(K=40)

图 2　MovieLens 数据集上,不同方法的推荐性能对比(K=20 和 K=40)

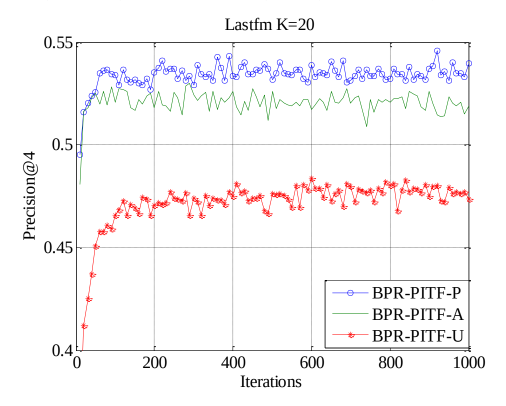
Fig.2　Comparison of recommended performance of different methods on MovieLens (K=20 and K=40)

3.4.2 收敛性能对比

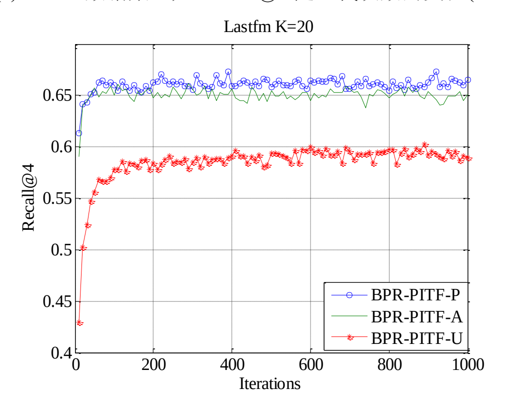
图 3 表示在 Lastfm 数据集（图 3（a）~（c））和 MovieLens 数据集（图 4（a）~（c））上，三个方法的推荐性能随迭代次数的变化。

从图中可以看出，两个数据集上，BPR-PITF-U 方法收敛最慢，且推荐的准确性最差；BPR-PITF-A 的收敛速度比较快，但推荐的准确性不如本文提出的 BPR-PITF-P 方法；BPR-PITF-P 方法的收敛能力介于 BPR-PITF-U 方法和 BPR-PITF-A 方法之间，但能够在较少迭代次数后取得相对稳定的结果。

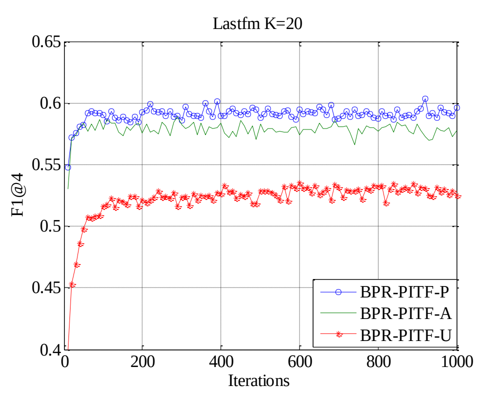
综上，本文提出的方法在保证一定的收敛能力的同时，取得相对较好的推荐准确性，较好地满足用户的个性化需求。



(a)Lastfm 数据集上,Precision@4 随迭代次数的变化(K=20)



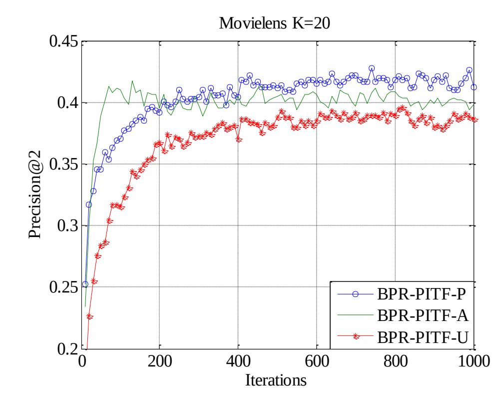
(b)Lastfm 数据集上,Recall@4 随迭代次数的变化(K=20)



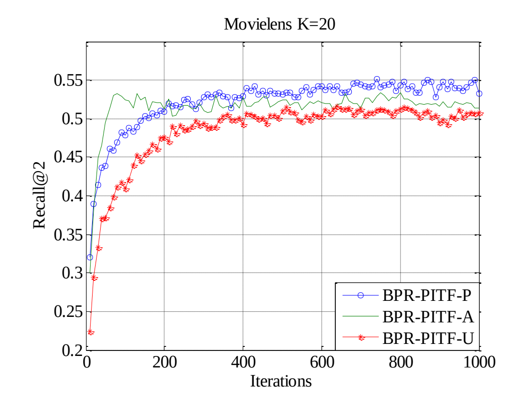
(c)Lastfm 数据集上,F1@4 随迭代次数的变化(K=20)

图 3　Lastfm 数据集上,不同方法的收敛性能对比(K=20)

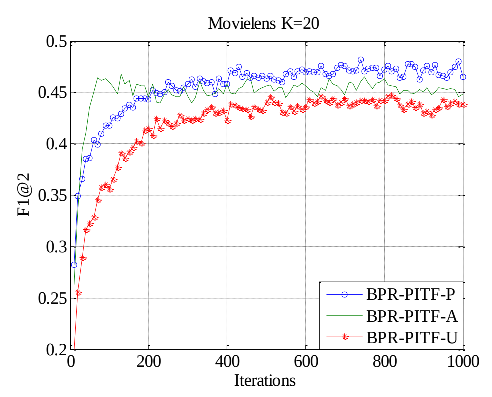
Fig.3　Comparison of convergence performance of different methods on Lastfm(K=20)



(a)MovieLens 数据集上,Precision@2 随迭代次数的变化(K=20)



(b)MovieLens 数据集上,Recall@2 随迭代次数的变化(K=20)



(c)MovieLens 数据集上,F1@2 随迭代次数的变化(K=20)

图 4　MovieLens 数据集上,不同方法的收敛性能对比(K=20)

Fig.4　Comparison of convergence performance of different methods on MovieLens(K=20)

4　结束语

个性化标签推荐系统能够为用户提供更加个性化的、高质量的标签来标注信息资源，但用户—资源—标签资源的稀疏性对现有推荐算法带来一定的挑战。针对该问题，提出通过结合用户和项目的上下文信息来挖掘大量、潜在有用的标签信息，改善对用户偏好关系的表达能力，缓解稀疏性的影响。而利用当前流行的个性化排序推荐框架和成对交互张量分解方法对偏好关系建模，提高标签排序推荐性能。对比实验表明，提出的方法在保证较快收敛速度的同时，提高了个性化标签的推荐准确性。

因此利用时间、用户社交关系等辅助数据来提高标签推荐性能是下一步的研究方向。

参考文献：

[1] Marinho L, Hotho A, Schmidt-Thieme L, *et al.* , Tag recommendations in social bookmarking systems [J]. Ai Communications, 2008, 21 (4): 231-247.

[2] Kim H N, Saddik A E. Personalized PageRank vectors for tag recommendations: inside FolkRank [C]// Proc of ACM Conference on



Recommender Systems. New York: ACM Press, 2011: 45-52.

[3] Ramezani M, Improving Graph-based Approaches for Personalized Tag Recommendation [J]. Journal of Emerging Technologies in Web Intelligence, 2011, 3 (2)

[4] Rendle S, Marinho L B, Nanopoulos A, *et al.* Learning optimal ranking with tensor factorization for tag recommendation [C]// Proc of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2009: 727-736.

[5] Rendle S, Schmidt-Thieme L. Pairwise interaction tensor factorization for personalized tag recommendation [C]// Proc of ACM International Conference on Web Search and Data Mining. New York: ACM Press, 2010: 81-90.

[6] Symeonidis P, Nanopoulos A, Manolopoulos Y, A Unified framework for providing recommendations in social tagging systems based on ternary semantic analysis [J]. IEEE Trans on Knowledge & Data Engineering, 2009, 22 (2): 179-192.

[7] Rendle S, Freudenthaler C. Improving pairwise learning for item recommendation from implicit feedback [C]// Procof the 7th ACM International Conference on Web Search and Data Mining: New York: ACM Press, 2014: 273-282.

[8] Krestel R, Fankhauser P, Personalized topic-based tag recommendation [J]. Neurocomputing, 2012, 76 (1): 61-70.

[9] 张斌，张引，高克宁，等，融合关系与内容分析的社会标签推荐 [J]. 软件学报, 2012, 23 (3): 476-488. (Zhang Bin, Zhang Yin, Gao Kening, *et al.* , Combining relation and content analysis for social tagging recommendation [J]. Journal of Software, 2012, 23 (3): 476 (488)

[10] Feng W, Wang J. Incorporating heterogeneous information for personalized tag recommendation in social tagging systems [C]// Proc of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2012: 1276-1284.

[11] Lops P, Gemmis M D, Semeraro G, *et al.* , Content-based and collaborative techniques for tag recommendation: an empirical evaluation [J]. Journal of Intelligent Information Systems, 2013, 40 (1): 41-61.

[12] 李贵，王爽，李征宇，等，基于时间加权三部图的分众分类标签推荐算法 [J]. 小型微型计算机系统, 2016, 37 (2): 269-274. (Li Gui, Wang Shuang, Li Zhengyu, *et al.* Folksonomy tag recommendation algorithm based on time-weighted tripartile graph [J]. Journal of Chinese computer system, 2016, 37 (2): 269-274.)

[13] Yu Hong, Zhou Bin, Deng Mingyao, *et al.* , Tag recommendation method in folksonomy based on user tagging status [J]. Journal of Intelligent Information Systems, 2017, (2): 1-22.

[14] Mahboob V A, Jalali M, Jahan M V, *et al.* , Swallow: resource and tag recommender system based on heat diffusion algorithm in social annotation systems [J]. Computational Intelligence, 2017, 33 (1): 99-118.

[15] Symeonidis P, Nanopoulos A, Manolopoulos Y, Tag recommendations based on tensor dimensionality reduction [C]// Proc of International Conference on Artificial Intelligence Applications & Innovations. 2009, 296: 331-340.

[16] Rendle S, Freudenthaler C, Gantner Z, *et al.* BPR: Bayesian personalized ranking from implicit feedback [C]// Proc of the 25th Conference on Uncertainty in Artificial Intelligence. AUAI Press, 2009: 452-461.

[17] Li Huayu, Ge Yong, Hong Richang, *et al.* Point-of-interest recommendations: Learning potential check-ins from friends [C]// Proc of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. New York: ACM Press, 2016: 975-984.

[18] Zhao Tong, McAuley J, King I. Leveraging social connections to improve personalized ranking for collaborative filtering [C]// Proceedings of the 23rd ACM international conference on information and knowledge management. New York: ACM Press, 2014: 261-270.